

Stochastic Minority on Graphs

Jean-Baptiste Rouquier^{1,2}, Damien Regnault^{1,2}, and Éric Thierry^{1,2,3}

¹ Université de Lyon, École Normale Supérieure de Lyon, LIP, France

² Institut des Systèmes Complexes Rhône-Alpes (IXXI), France

³ CNRS, LIAFA, Université Paris 7, France

Abstract. Cellular automata have been mainly studied on very regular graphs carrying the vertices (like lines or grids) and under synchronous dynamics (all vertices update simultaneously). In this paper, we study how the asynchronism and the graph act upon the dynamics of the classical Minority rule. Minority has been well-studied for synchronous updates and is thus a reasonable choice to begin with. Yet, beyond its apparent simplicity, this rule yields complex behaviors when asynchronism is introduced. We investigate the transitory part as well as the asymptotic behavior of the dynamics under full asynchronism (also called sequential: only one random vertex updates at each time step) for several types of graphs. Such a comparative study is a first step in understanding how the asynchronous dynamics is linked to the topology (the graph).

Previous analyses on the grid [1,2] have observed that Minority seems to induce fast stabilization. We investigate here this property on arbitrary graphs using tools such as energy, particles and random walks. We show that the worst case convergence time is, in fact, strongly dependent on the topology. In particular, we observe that the case of trees is non trivial.

1 Introduction

In this paper, we investigate the random process Minority: each vertex of a graph is characterized by a state 0 or 1. At each time step (the time is discrete), one random subset of vertices is drawn. These vertices are updated and switch to the minority state in their neighborhood.

Similar random processes appeared in the literature and concern different fields of applications. For example, several studies focus on the emergence of cooperation in the iterated prisoner's dilemma game [3,4,5]: used as a simple social model, this field helped determine which ingredients are needed to foster the emergence of cooperative behaviors. The rock-paper-scissors game was used to model evolution of colonies of bacteria [6]. In physics, the Ising model (a kind of Majority rule) was introduced to study ferromagnetism (original paper is [7], see for instance Chapter 10 of [8] for a quick introduction): states represent the orientation of spin. Anti-ferromagnetism is studied by using a kind of Minority rule [9]. Recent works [10,11] use stochastic Minority to model the formation of quasi-crystals.

Initially, our interest in studying Minority process comes from the field of cellular automata (CA). CA can be seen both as a model of computation with massive parallelism and as a model for complex systems in nature. They have been studied with various fields of

applications like parallel/distributed computing, physics, biology or social sciences. Most of the work regarding CA assumes that their dynamics is deterministic and synchronous (all vertices update simultaneously) and that the graph is very regular (usually a line or a 2D or 3D grid). Such assumptions can be questioned with regard to the applications and the real life constraints. Dynamics where those assumptions are perturbed have been far less studied and their analysis is very challenging. Here is a non-exhaustive list of related works about CA in literature, the models are *stochastic CA* since the perturbations are usually introduced as stochastic processes:

- Perturbation of the updating rule: resilience to random errors [12,13,14,15,16,17], mean field analysis of general Markovian rules [18].
- Perturbation of the synchronism (i.e. of the updating scheme): empirical studies about resilience to asynchronism [19,20,21,22,23], mathematical analysis of some 1D CA under full asynchronism (only one random vertex updates at each time step) or under α -asynchronism (each vertex updates independently with probability α) [24,25,26].
- Perturbation of the graph (the topology of cells): empirical studies [27,28,29], gene regulatory networks [30,31].

Previous analyses focus on the effects of asynchronism on 2D Minority with Von Neumann neighborhood [1] and Moore neighborhood [2]. In this paper, we choose to investigate how the graph acts upon the dynamics under asynchronous updates. (Note that this is a special case of Interacting Particle Systems [32].) We focus on *Stochastic Minority* where the Minority rule applies to two possible states $\{0, 1\}$ and under full asynchronism (at each time step, only one random vertex is updated with the uniform distribution). This simple rule already exhibits a surprisingly rich behavior as observed in [1,2] where it is studied for vertices assembled into a torus. Such behaviors may appear because Minority is a CA with *negative* feedback. The evolution of CA with *positive* feedback can be described by a bounded decreasing function over time [33]. Thus, the difficulty of analyzing the Minority rule (negative feedback) must not be confounded with the difficulty of analyzing the Majority rule (positive feedback). Some related stochastic models like Ising models or Hopfield nets have been studied under asynchronous dynamics (e.g. our model of asynchronism corresponds to the limit when temperature goes to 0 in the Ising model). These models are acknowledged to be harder to analyze when it comes to arbitrary graphs [34,9] or negative feedbacks [35].

Let us stress that we mainly study stochastic Minority on several kinds of graphs: trees, cliques, bipartite graphs and compare them. General results are not precise enough to describe its behavior in the present study. One of the aim of this paper is to show that Minority's behavior highly depends on the topology of the graph. Our paper focus on particular classes of graphs. It is a first step and the reasonings may prove to be helpful to study future applications of Minority (as in [10,11]) and our results complete some previous results about Minority [1,2].

Here is a list of previous claims about Stochastic Minority in the literature, as well as some insights provided by the present paper.

Short introduction of stochastic Minority behaviors: Figure 1 shows Minority on a 2D grid under three different dynamics:

- the α -*synchronous* dynamics where each cell has a probability α to be updated independently from the others at each time step;
- the *synchronous* dynamics where all cells are updated at each time step (α -asynchronous dynamics for $\alpha = 1$);
- the *fully asynchronous* dynamics where only one cell, randomly and uniformly chosen, is updated at each time step (it can be regarded as, and often behaves as the limit of the α -asynchronous dynamics when α tend to 0 [25,36]).

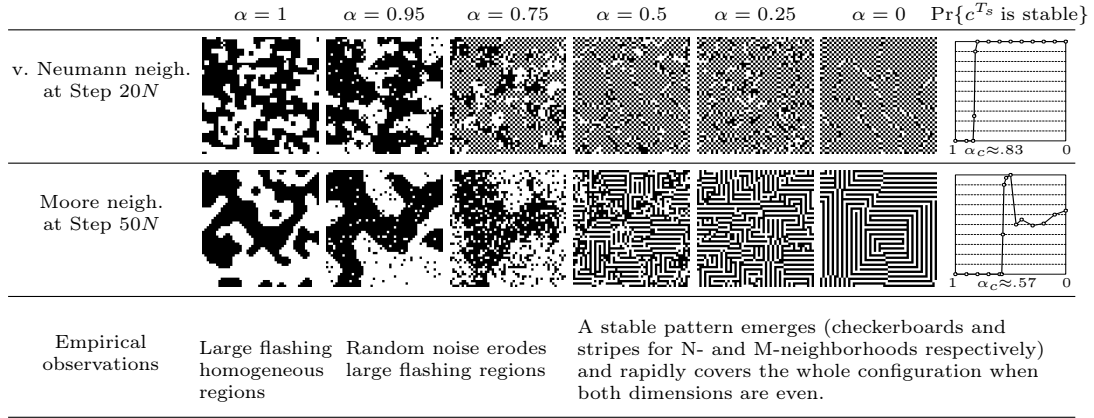


Fig. 1. Stochastic Minority under different α -asynchronous dynamics with $N_{50} = 50 \times 50$ cells arranged in a $2D$ grid with periodic boundary condition (i.e. torus). The last column gives, for $\alpha \in [0,1]$, the empirical probability that an initial random configuration converges to a stable configuration before time step $t_s \cdot N_{50}$ where $t_s = 1000$ and $t_s = 2000$ for von Neumann and Moore neighborhood respectively. The fully asynchronous dynamics is abusively designed by $\alpha = 0$.

Depending on the value α , Minority can exhibit two different behaviors. Experimentally, these phenomenon can be observed as a phase transition depending on α on the convergence time (see figure 1):

- When α is almost 1, there is a big homogeneous *flashing* background with some random noise. By flashing, we mean that all the cells of the background are not in their minority state and since α is almost 1, they switch their states at almost each time step. The few cells which are not updated create some random noise. Experimentally, the dynamics never reaches a stable configuration, but a highly improbable series of updated may lead to a stable configuration.
- When α is almost 0, different regions made of checkerboard patterns (von Neumann neighborhood) or stripes (Moore neighborhood) quickly appear. The only cells which may switch their state are along the borders between these regions. These borders evolve and they eventually stabilize. Experimentally, the dynamics reaches a stable configuration in polynomial time according to the size of the configuration.

No results were previously known for the α -asynchronous dynamics. In section 4.3, we study this phase transition in $1D$ and show that it is linked to directed percolation. The proof of this result (theorem 34) is short because most of the technical aspects were previously done in [37] on a different automaton.

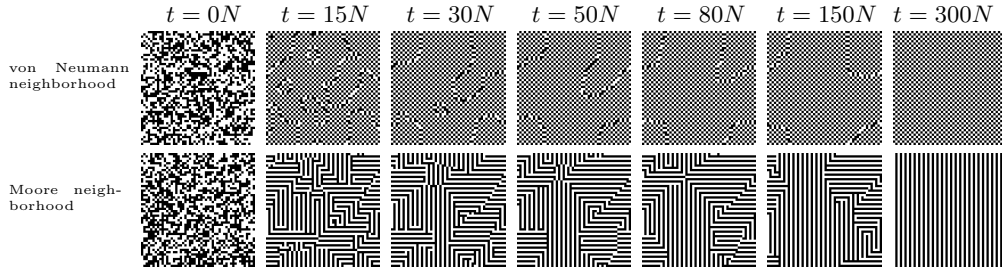


Fig. 2. Minority under fully asynchronous dynamics on $2D$ grids with periodic boundary condition, von Neumann and Moore neighborhood [1,2].

The previous studies [1,2] focuses on the fully asynchronous dynamics which is similar to the α -asynchronous dynamics when α is almost 0 (see Figure 2). These papers analyse the start and the end of a classical execution on stochastic Minority on a $2D$ grid with von Neumann and Moore neighborhood. Indeed, they had to study separately the formation of the regions (beginning) and the evolution of the borders (ending).

Local interactions: under fully asynchronous dynamics, patterns, which depend on the topology of the graph, quickly appear. In theorems 5 and 7 of [1] and theorem 9 of [2], the authors show that this phenomenon is due to *local interactions* and occurs in polynomial time according to the size of the configuration. When these papers were written, the authors did not need to be more precise but they conjectured that these patterns appear in linear time. Here by studying cliques (where long range interactions do not exist), we show that the dynamics stabilizes in linear time. This result supports the previous conjecture.

Long range interactions: the evolution of borders between different regions often implies interactions between cells which could be arbitrarily far away in the graph. This long range interactions are harder to analyze and major differences appear between the von Neumann and Moore neighborhood on the $2D$ grid. This remark leads us to analyze different topologies. Here we solve a conjecture made in [1]. We exhibit biased trees where Minority behaves as a biased random walk and need an exponential time to converge toward a stable configuration. In the previously considered topologies, Minority always converges in polynomial time under fully asynchronous dynamics.

Capacity of simulation: one important aspect of Minority is the diversity of phenomena embedded in this rule. On α -asynchronous dynamics, there is a phase transition between two different behaviors. In section 4.3, we establish a link between this phase transition and *directed percolation*. Previous studies [1,2] have shown that the fully asynchronous dynamics occurs in two steps. We show here (Section 4.1) that the first step acts as a *coupon collector* on cliques. When long range interactions occur, the dynamics may behave as competition

between $2D$ regions, different kinds of *random walks*. Other behaviors may be encoded on stochastic Minority. It is surprising that a simple rule may produce such different behaviors from random configurations on regular topologies.

Bipartite graphs: one aim of this paper is to generalize tools previously designed to study Minority on grid. Basic tools may be generalized to any graph but we realized that advanced tools can only be generalized to bipartite graphs. This leads to an explanation between the difference of behaviors previously observed on $2D$ grids. Note that even if our tools are helpful to analyze bipartite graphs, various and complicated behaviors may already appears in this class of graphs.

2 Model

In this paper we consider Stochastic Minority on arbitrary undirected graphs.

Definition 1 (Configuration). Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a finite undirected graph with vertices \mathcal{V} and edges \mathcal{E} . $\mathcal{Q} = \{0, 1\}$ is the set of states (0 stands for white and 1 stands for black). The vertices are also called cells and $N := |\mathcal{V}|$ denotes their number. The neighborhood \mathcal{N}_i of a vertex i is the set of its adjacent vertices (including itself). A configuration is a function $c : \mathcal{V} \rightarrow \mathcal{Q}$ (c_i denotes the state of vertex i in configuration c).

Definition 2 (Stochastic Minority). We consider the following dynamics δ that associates with each configuration c a random configuration c' obtained as follows: a vertex $i \in \mathcal{V}$ is selected uniformly at random (we say that vertex i is fired) and its state is updated to the minority state among its neighborhood (no change in case of equality), while all the other vertices remain in their current state. Formally:

$$(\delta(c))_i = \begin{cases} 1 & \text{if } \sum_{j \in \mathcal{N}_i} c_j < \frac{|\mathcal{N}_i|}{2} \quad \text{or} \quad \sum_{j \in \mathcal{N}_i} c_j = \frac{|\mathcal{N}_i|}{2} \text{ and } c_i = 1 \\ 0 & \text{if } \sum_{j \in \mathcal{N}_i} c_j > \frac{|\mathcal{N}_i|}{2} \quad \text{or} \quad \sum_{j \in \mathcal{N}_i} c_j = \frac{|\mathcal{N}_i|}{2} \text{ and } c_i = 0 \end{cases}$$

and $(\delta(c))_k = c_k$ for all $k \neq i$. In a configuration, a vertex is said to be active if its state changes when the vertex is fired. The random variable c^t denotes the configuration obtained from an initial configuration c , after t steps of the dynamics: $c^0 = c$ and $c^t = \delta^t(c)$ for all $t \geq 1$. (The notation $\delta^t(c)$ means $\delta \circ \dots \circ \delta(c)$, t times).

Definition 3 (Attractors and limit set). For the dynamics induced by δ , a set of configurations A is an attractor if for all $c, c' \in A$, the time to reach c' starting from c is finite almost surely. In the transition graph where vertices are all the possible configurations and arcs (c, c') satisfy $\mathbb{P}(\delta(c) = c') > 0$, attractors are the strongly connected components with no arc leaving the component. The union of all attractors is denoted \mathcal{A} and called the limit set.

Definition 4 (Convergence and hitting time). We say that the dynamics δ converges from an initial configuration c^0 to an attractor A (resp. the limit set \mathcal{A}) if the random variable $T = \min \{t \mid c^t \in A\}$ (resp. $T = \min \{t \mid c^t \in \mathcal{A}\}$) is almost surely finite.

The variable T is a hitting time.

Since we only consider finite graphs, the dynamics δ always converges from any initial configuration to \mathcal{A} , and T is well-defined (an exception is section 4.3, where we consider an infinite graph and discuss the convergence to an attractor).

A hitting time T is defined for a given graph and a given initial configuration. We are interested in the worst case (i.e. largest hitting time) among all possible initial configurations and among all graphs of a given class of graphs.

3 Tools

3.1 Energy, Potential

As in the Ising model [9] or in Hopfield networks [35], we define a natural global parameter similar to an energy: it counts the number of interactions between neighboring vertices in the same state. This parameter will provide key insights into the evolution of the system.

Definition 5 (Potential). *The potential v_i of vertex i is the number of its neighbors (including itself) in the same state as itself. If $v_i \leq \frac{|\mathcal{N}_i|}{2}$ then the vertex is in the minority state and is thus inactive; whereas, if $v_i > \frac{|\mathcal{N}_i|}{2}$ then the vertex is active. A configuration c is stable if and only if for all vertex $i \in \mathcal{V}$, $v_i \leq \frac{|\mathcal{N}_i|}{2}$.*

Definition 6 (Energy). *The energy of configuration c is $E(c) = \sum_{i \in \mathcal{V}} (v_i - 1)$.*

The energy of a configuration is always non-negative. There are configurations of energy 0 if and only if \mathcal{G} is bipartite: those stable configurations are the 2-colorings of \mathcal{G} . More generally, we have:

Proposition 7 (Energy bounds). *The energy E satisfies $2|\mathcal{E}| - 2C_{\max} \leq E \leq 2|\mathcal{E}|$, where C_{\max} is the maximum number of edges in a cut of \mathcal{G} .*

Proof. The bounds are direct consequences of the definitions: for any configuration, $E = 2|\mathcal{E}| - 2|C|$ where C is the cut $\{\{i, j\} \in \mathcal{E} \mid c_i = 0 \text{ and } c_j = 1\}$.

As a consequence, computing the minimum energy for arbitrary graphs is NP-hard: it is equivalent to computing MAXCUT.

Lemma 8 (Energy is non-increasing). *The energy is a non-increasing function of time and decreases each time a vertex i with potential strictly larger than $\frac{|\mathcal{N}_i|}{2}$ fires.*

Proof. When an active vertex i of potential v_i fires, its potential becomes $|\mathcal{N}_i| - v_i + 1$, and the energy of the configuration becomes $E + 2|\mathcal{N}_i| - 4v_i + 2$. \square

Corollary 9. *A configuration c belongs to the limit set if and only if no sequence of updates would lead the energy to decrease, i.e. if and only if $\forall t, \mathbb{P}(E(c^t) > E(\delta(c^t)) \mid c^0 = c) = 0$.*

Proof. If the energy decreases when updating c to c' , then c will never be reached again (because energy is non-increasing). Reciprocally, any update that keeps the energy constant is reversible: the fired vertex can be fired again to get back to the previous configuration. \square

Remark 10. Since firing a vertex of odd degree makes the energy decrease, such vertices are inactive in the limit set.

Definition 11 (Particle). Let c be a configuration on $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, an edge $\{i, j\}$ holds a particle if $c_i = c_j$. A configuration is fully characterized (up to the black/white symmetry) by its set of particles located at $\mathcal{P} \subseteq \mathcal{E}$.

(Note that the converse proposition “any subset $\mathcal{P} \subseteq \mathcal{E}$ corresponds to a configuration” is true if and only if the graph is a tree).

The energy of a configuration is clearly equal to twice its number of particles. With the particle point of view, when firing a vertex i of degree $\deg(i) = |\mathcal{N}_i| - 1$, if the number of incident edges holding a particle is a least $\frac{\deg(i)}{2}$, these particles disappear but new particles appear on the incident edges (if any) which had no particle (as illustrated on Fig. 3). Otherwise the particles do not move.

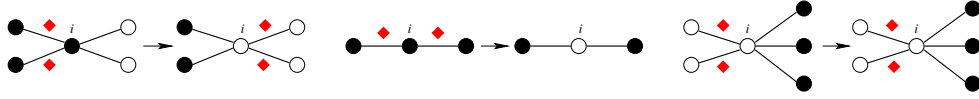


Fig. 3. Transfers of particles (diamonds) when firing vertex i .

Switching between the coloring and the particle points of view may simplify the description of the configurations and the dynamics, e.g. when the energy is low and the dynamics comes to random walks of a few particles.

3.2 Bipartite Graphs

A graph is bipartite when its vertices can be partitioned into two sets such that every edge goes from one set to the other (or equivalently, when it is 2-colorable). Bipartite graphs allows us to use another, easier point of view (the dual configuration) and to easily determine if a configuration is in the limit set.

Dual configuration We now introduce dual configurations as in [1] (section 3.2), and their dual rule to facilitate the study of the dynamics on trees. In this dual dynamics equivalent to Stochastic Minority, the stable configurations of minimum energy are the two configurations all black and all white, and the regions which compete are all white versus all black subtrees.

The dual rule is almost a majority rule, but in case of equality among the neighbors of a vertex, the state of this vertex is flipped each time it is updated. This “instability” prevents many results about majority rules to apply to our case.

Definition 12 (Dual configurations). Consider a graph \mathcal{G} and fix a vertex r (the “root”). For any configuration c on \mathcal{G} , its dual configuration \hat{c} is defined as $\hat{c}_i = c_i$ if h_i is even and $\hat{c}_i = 1 - c_i$ if h_i is odd, where h_i is the distance from r to i (see Fig. 4). The mapping $c \mapsto \hat{c}$ is a bijection on the set of all the configurations; more precisely $\hat{\hat{c}} = c$.

An equivalent definition consists in making a XOR with the 2-coloring of \mathcal{G} such that r is white. The duals of the configurations of minimum energy 0 are the configuration where all vertices are black or all vertices are white.

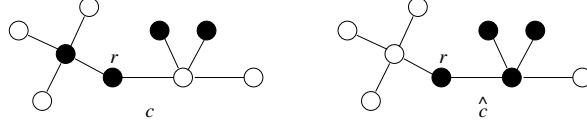


Fig. 4. A configuration c and its dual configuration \hat{c} (with regard to root r).

Proposition 13 (Dual dynamics). *Consider a sequence (c^t) for the Stochastic Minority dynamics δ and the sequence (\hat{c}^t) of the dual configurations, and define the dual dynamics $\hat{\delta}$ as $\hat{\delta}(\hat{c}) = \widehat{\delta(c)}$ so that $\hat{c}^{t+1} = \hat{\delta}(\hat{c}^t)$. Then the dynamics $\hat{\delta}$ is also a stochastic CA. It associates with each configuration \hat{c} a random configuration \hat{c}' by updating one random vertex i uniformly with the rule which selects the majority state in the neighborhood of i excluding itself (in case of equality its state changes):*

$$\hat{c}'_i = \begin{cases} 1 & \text{if } \sum_{j \in \mathcal{N}_i \setminus \{i\}} \hat{c}_j > \frac{|\mathcal{N}_i| - 1}{2} \text{ or } (\sum_{j \in \mathcal{N}_i \setminus \{i\}} \hat{c}_j = \frac{|\mathcal{N}_i| - 1}{2} \text{ and } \hat{c}_i = 0) \\ 0 & \text{if } \sum_{j \in \mathcal{N}_i \setminus \{i\}} \hat{c}_j < \frac{|\mathcal{N}_i| - 1}{2} \text{ or } (\sum_{j \in \mathcal{N}_i \setminus \{i\}} \hat{c}_j = \frac{|\mathcal{N}_i| - 1}{2} \text{ and } \hat{c}_i = 1) \end{cases}$$

By construction, the dual sequences (c^t) and (\hat{c}^t) as well as their corresponding dynamics δ and $\hat{\delta}$ are stochastically coupled (see [38]) by firing the same random vertex at each time step.

Definition 14 (Dual potential & Energy). *The dual potential \hat{v}_i of vertex i is the number of its neighbors (excluding itself) in a different state than itself. If $\hat{v}_i < \frac{|\mathcal{N}_i| - 1}{2}$ then the vertex is in the majority state and is thus inactive; whereas, if $\hat{v}_i \geq \frac{|\mathcal{N}_i| - 1}{2}$ then the vertex is active. The dual energy \hat{E} is the sum of the dual potentials over all the vertices.*

Given a configuration c and its dual \hat{c} , the potential of any vertex i in c is equal to the dual potential of vertex i in \hat{c} plus 1. Thus the dual energy of \hat{c} is exactly the energy of c .

Theorem 15. *Consider a configuration c and its dual \hat{c} . Then, if the same vertex fires, $\widehat{\delta(c)} = \hat{\delta}(\hat{c})$. We thus have the following commutative diagram:*

$$\begin{array}{ccc} \text{original} & & \\ & c \xrightarrow{\delta} c' & \\ & \updownarrow & \updownarrow \\ & \hat{c} \xrightarrow{\hat{\delta}} \hat{c}' & \\ \text{dual} & & \end{array}$$

Proof. Consider a configuration c and its dual \hat{c} . A vertex c_i of c is active if and only if $v_i \leq \frac{N_i}{2}$. Since $v_i = \hat{v}_i + 1$ and \hat{v}_i is an integer, a vertex c_i of c is active if and only if $\hat{v}_i < \frac{N_i-1}{2}$: that is to say vertex \hat{c}_i of \hat{c} is active. According to definition 12, if the same vertex fires and a vertex of c is active if and only if the corresponding vertex of \hat{c} is active then $\widehat{\delta(c)} = \hat{\delta}(\hat{c})$. \square

Distance to a Stable Configuration In this section we describe an algorithm (algorithm 1) that gives a sequence of updates that leads to the limit set. It is then easy to test for the limit set: the input configuration belongs to the limit set if and only if the energy is the same between the input and output configurations.

Fact 16 *An attractor A decomposes the graph into three sets of vertices:*

1. *the vertices that are in the state 0 for every configuration of A ;*
2. *the vertices that are in the state 1 for every configuration of A ;*
3. *the vertices that can be either in the state 0 or 1, depending on the configuration in A .*

Algorithm 1: Membership to the limit set: check that $\widehat{E}(c') = \widehat{E}(c)$.

Input: A configuration c .

1 **while** *There is an active white vertex i* **do** Fire i (i becomes black)

2 **while** *There is an active black vertex i* **do** Fire i

3 **while** *There is an active white vertex i* **do** Fire i

Output: The configuration c' at the end of phase 2.

Proposition 17. *The configuration c' returned by algorithm 1 is in the limit set.*

Corollary 18. *The input configuration c is in the limit set if and only if the energy has not decreased during execution of the algorithm.*

Proof. We first prove that the vertices in state 0 (white) at the end of phase 1 of the algorithm cannot switch to state 1, whatever the sequence of updates, i.e. they are in Case 1 of Fact 16. Indeed, assume instead that there exists a vertex i and sequence of configurations c^1, c^2, \dots, c^k such that

- c^1 is the configuration at the end of phase 1;
- each configuration is the result of firing one vertex in the previous configuration;
- $c_i^1 = 0$ and $c_i^k = 1$.

Let c^ℓ be the configuration just before the first update of this sequence that fires an active vertex j in state 0 (there exists one since at least i will be fired in this sequence). But j must have been already active in c^1 , since it had at least as many neighbors in state 1 as in c^ℓ (this is a monotonicity argument). The algorithm thus would not have exited the first “while” loop, which is a contradiction.

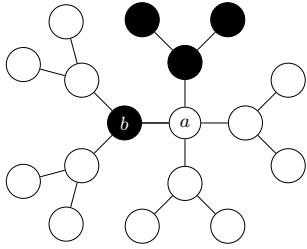
Same holds for the vertices in state 0 at the end of phase 3. Also, by symmetry, the vertices in state 1 at the end of phase 2 are in Case 2 of Fact 16.

Finally, since the remaining vertices were white at the end of phase 2, they can be made white by a sequence of updates starting from the configuration c^1 (the configuration at the end of phase 1). By monotonicity, they can be made white by the same sequence of updates, starting from any configuration reachable from c^1 . This means that the configuration where all those remaining vertices are white is always reachable, and is thus in the attractor. This configuration is the one at the end of phase 2. Which concludes the proof.

By symmetry, those remaining vertices can be made black from the configuration at the end of phase 2, which is in the attractor, so they are in Case 3 of Fact 16. \square

Termination of Algorithm 1 is clear: phase 1 increases the number of vertices in state 1 at each iteration and is thus completed in $O(N)$ iterations. The same remark applies to the other phases.

Three phases are necessary to correctly classify the vertices into the three cases of fact 16, as shown on Fig. 5. Two phases are enough to reach the limit set.



Starting from this configuration, phase 1 of the algorithm makes the vertex a become black. Then, phase 2 makes its left neighbor b become white, as well as a .

b is now definitely white (Case 1 of Fact 16), but was not identified as such at the end of phase 1. Thus, a third phase is necessary with this algorithm to identify b .

No other vertex will be active during the execution of the algorithm.

Fig. 5. Three phases are necessary.

Stable Configurations Stable configurations on trees are characterized in the appendix, Section A.

3.3 Non Bipartite Graphs

It seems that the only in-depth study of a non-bipartite graph is the analysis of the $2D$ grid with Moore neighborhood and periodic boundary condition [2]. This study is harder and more complicated than any bipartite graphs considered so far. This complexity appears in the structure of the stable configurations. For example, Figure 6 shows a stable configuration, illustrating many ways in which the four different stable patterns can be intricately. Perturbing such stable configurations leads to original random walks (see Figure 7). Particles appear and move along the borders. They deform the border when they move. When two particles collide, they disappear. When borders are too perturbed, the whole structure of the stable configuration collapses. As opposed to the bipartite case, we are currently not able to analyze such dynamics and compute for any configuration a sequence of updates leading to a stable configuration.

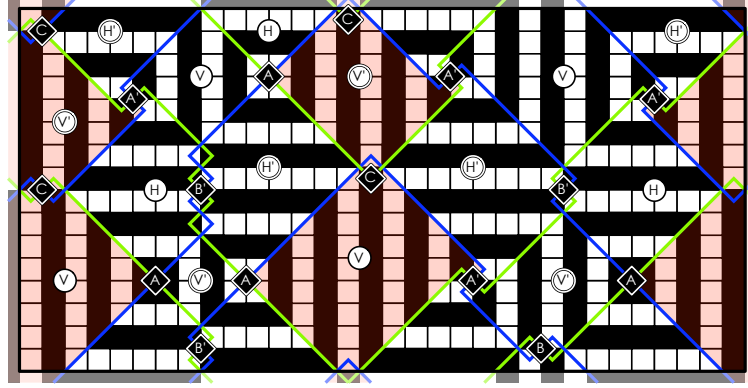
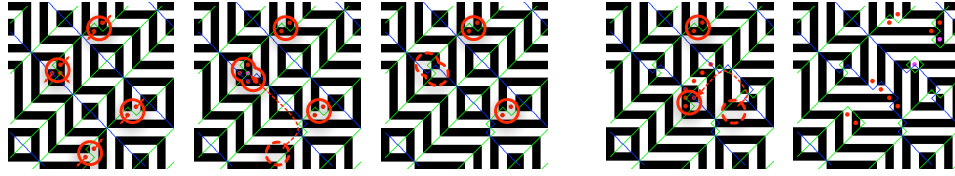


Fig. 6. A complicated 16×32 stable configuration. There are four stable patterns (V,V',H,H') and different types of junction between the different regions(A, A', B, B' and C).



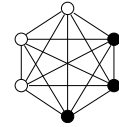
7.a – A sequence of updates in a configuration 7.b – A sequence of updates where the rails starting with 4 particles where two of them move cannot sustain the perturbations due to the along rails and ultimately vanish after colliding movements of the particles: at some point, rails get to close with each other, new active cells appear, and part of the rail network collapses.

Fig. 7. Some examples of the complex behavior of particles in a 20×20 configuration.

4 Behaviors

4.1 Decreasing Energy (Clique, cycle and paths)

We prove in this part that Stochastic Minority on cliques behaves as a *coupon collector* (see [39], page 210). This easy result implies a fast convergence to the limit set.



Theorem 19. *Stochastic Minority on cliques hits the limit set after $O(N \log N)$ steps on expectation. If N is even, the $\binom{N}{\frac{N}{2}}$ attractors are the half black and half white configurations, each one stable. If N is odd, the only attractor is the set of $2\binom{N-1}{\frac{N-1}{2}}$ configurations having one more (resp less) black than white vertices.*

Proof. Let n_b be the number of black vertices of a configuration. Since the neighborhood of a vertex is \mathcal{V} , the potential of a black vertex is n_b . If $n_b > \frac{N+1}{2}$ (resp. $n_b < \frac{N-1}{2}$) then firing a black (resp. white) vertex decreases the energy and the configuration is not in \mathcal{A} . Thus a configuration in \mathcal{A} must have $\frac{N-1}{2} \leq n_b \leq \frac{N+1}{2}$. Consider such a configuration:

- If N is even then all vertices have potential $\frac{N}{2}$ and these $\binom{N}{\frac{N}{2}}$ configurations are stable.
- If N is odd we call C_b (resp. C_w) the set of configurations where $n_b = \frac{N+1}{2}$ (resp. $n_b = \frac{N-1}{2}$). White (resp. black) vertices of a configuration in C_b (resp. C_w) are inactive and black (resp. white) vertices are active, firing one of them leads with constant energy to a configuration of C_w (resp. C_b). Thus from any configuration of $C_w \cup C_b$, there is no sequence of updates that causes a drop of energy and $C_w \cup C_b = \mathcal{A}$.

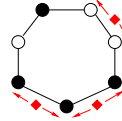
Now, we prove that \mathcal{A} is made of only one attractor. Let $d(c, c') := |\{i \mid c_i \neq c'_i\}|$ be the distance between two configurations. Consider in \mathcal{A} two configurations $c \neq c'$. By symmetry we can assume $c \in C_b$. Since c has at least as many black vertices as c' , there is a vertex i black in c and white in c' . Firing i decreases the distance. Iterating this argument, one finds a path from c to c' in \mathcal{A} .

Now consider a configuration where $n_b > \frac{N+1}{2}$. As long as the configuration does not belong to \mathcal{A} , the white vertices are inactive. When $\lceil n_b - \frac{N+1}{2} \rceil$ black vertices have fired, the configuration is in \mathcal{A} . At each time step there is a probability $\frac{n_b}{N}$ to fire a black vertex. This kind of dynamics is known as coupon collector and $T = O(N \log N)$. \square

Moreover, it is easy to see that in the odd N case, the attractor has a simple structure: this is a bipartite graph, composed of configurations with $\frac{N-1}{2}$ white vertices on one side, and configurations with $\frac{N+1}{2}$ white vertices on the other.

Cycles and paths Cycles and paths forms the class of connected graphs of maximum degree 2.

On cycles and paths, the particle point of view is convenient and one can prove that Stochastic Minority behaves as *random walks of annihilating particles* on a discrete ring (see [24,40]). On the right, the particles are the diamonds.



Theorem 20. *Stochastic Minority on cycles and paths hits the limit set after $O(N^3)$ steps on expectation. If N is even, the two attractors are the 2-colorings of the cycle. If N is odd, the single attractor is a cycle in the transition graph composed of all the configurations with only one particle.*

Proof. The movement of particles provide a nice framework for the analysis. Firing a vertex incident to a particle either attract the particle on the next edge if it is free, or annihilate both particles on each side of the vertex. Consequently, the dynamics boils down to the analysis of random walks of annihilating particles on a discrete ring. If the number N of vertices is even (resp. odd), any configuration has necessarily an even (resp. odd) number of particles. This number decreases by annihilations until there is no (resp. only one) particle.

The attractor is reached at this point since the energy cannot decrease further. This proof can be easily adapted to graphs which are paths.

To bound the expected hitting time of the limit set, associate with each configuration c^t a weight X_t which is the maximum distance between two consecutive particles if there are at least two particles, or N if there is only one particle, or $N + 1$ if there is no particle. For all t , $X_t \in \{1, \dots, N + 1\}$ and c_t belongs to the limit set if and only if $X_t \in \{N, N + 1\}$. Let $\Delta X_{t+1} = X_{t+1} - X_t$. One can check that $\mathbb{E}(\Delta X_{t+1} | c_t = c) \geq 0$ for any configuration c . Moreover $\mathbb{E}((\Delta X_{t+1})^2 | c_t = c) \geq 3/N$ for any configuration c not in the limit set. Consequently $X_t^2 - 3t/N$ is a sub-martingale and we can apply the Stopping Time Theorem to the stopping time $T = \min \{t \geq 0 \mid X_t \in \{N, N + 1\}\}$. It gives $\mathbb{E}(T) = O(N^3)$ which is thus an upper bound on the expected hitting time of the limit set.

This proof applies with no modification to the hitting time on graphs which are paths. \square

4.2 Long Range Interaction and Exponential Convergence (Trees)

In this part, we introduce biased trees (Definition 24 and Figure 9) such that the dynamics \hat{c} converges in exponential time on this topology (Theorem 31). Vertices of biased trees have degree at most 4. In fact, biased trees simulate biased random walks (Definition 21) which converge in exponential time. Biased trees are created from small trees called widgets (Definition 23 and Figure 8) arranged on a line. Except at the ends, this line of widgets is made of “gates”. According to the configuration, these gates are either locked, unlocked or stable (Definition 25). On a correct configuration (Definition 26), the line of gates is split into two regions: all gates on the left side are stable and all gates on the right side are unstable (locked or unlocked). In a correct configuration, three different events may be triggered with the same probability $1/N$ (Fact 27 and Corollary 30):

- the rightmost stable gate becomes an unlocked gate;
- the leftmost unstable gate becomes stable if it is unlocked;
- the leftmost unstable gate is switched from locked to unlocked or the contrary.

Thus stable gates tend to disappear. This dynamics will ultimately converge to the stable configuration \hat{c}_f (Definition 28). To reach this configuration all gates must be stable. Thus it takes an exponential time for the dynamics \hat{c} to converge on a biased tree with an initial correct configuration.

Definition 21 (Biased random walks). A Biased Random Walk is a sequence of random variables $(X_i)_{i \geq 0}$ defined on $\{0, \dots, n\}$ such that for all $i \geq 0$:

- $\mathbb{P}(X_{i+1} = 1 \mid X_i = 0) = 1$ (reflecting barrier at 0).
- $\mathbb{P}(X_{i+1} = n \mid X_i = n) = 1$ (absorbing barrier at n).
- $\exists a, b \in \mathbb{R}_+ \quad \forall x \in \{1, \dots, n-1\} \quad \begin{cases} \mathbb{P}(X_{i+1} = x-1 \mid X_i = x) \\ + \mathbb{P}(X_{i+1} = x+1 \mid X_i = x) = 1 \\ 0 < a < \mathbb{P}(X_{i+1} = x+1 \mid X_i = x) < b < 1/2 \end{cases}$

Theorem 22. Let $T := \min \{i \geq 0 \mid X_i = n\}$ be the absorption time at n and for all $0 \leq k \leq n$, let $\mathbb{E}_k(T) := \mathbb{E}(T \mid X_0 = k)$ be its expectation starting from k . Then

$$\theta_k(b) \leq \mathbb{E}_k(T) \leq \theta_k(a)$$

$$\text{where } \theta_k(p) = \frac{2p(1-p)}{(1-2p)^2} \left(\left(\frac{1-p}{p} \right)^n - \left(\frac{1-p}{p} \right)^k \right) - \frac{n-k}{1-2p}.$$

This theorem is a direct consequence of classical analysis of random walks on $\{0, \dots, n\}$ where

$$\begin{aligned} & - \mathbb{P}(X_{i+1} = 1 \mid X_i = 0) = 1 \\ & - \mathbb{P}(X_{i+1} = n \mid X_i = n) = 1 \\ & - \forall x \in \{1, \dots, n-1\} \quad \begin{cases} \mathbb{P}(X_{i+1} = x+1 \mid X_i = x) = p \\ \mathbb{P}(X_{i+1} = x-1 \mid X_i = x) = q \end{cases}, \text{ with } p+q=1. \end{aligned}$$

Solving the following system of equations [39]:

$$\begin{cases} \mathbb{E}_k(T) = p(1 + \mathbb{E}_{k+1}(T)) + (1-p)(1 + \mathbb{E}_{k-1}(T)) & \forall k \in \{0, \dots, n\} \\ \mathbb{E}_n(T) = 0 \\ \mathbb{E}_0(T) = 1 + \mathbb{E}_1(T) \end{cases}$$

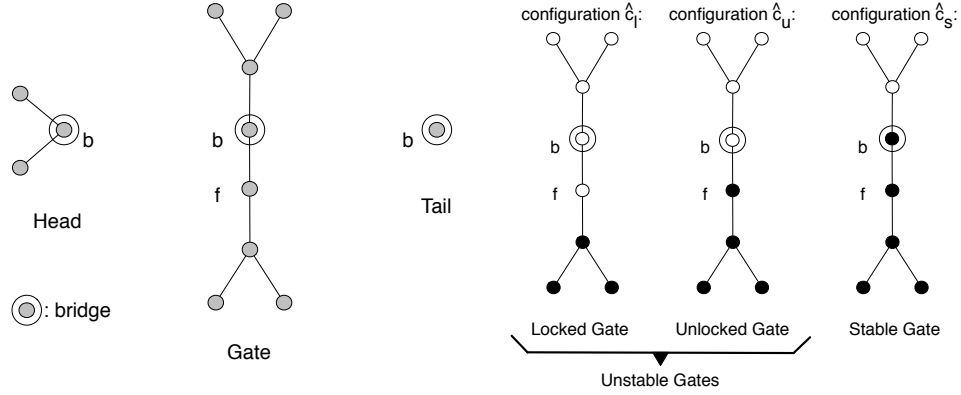
$$\text{one gets } \mathbb{E}_k(T) = \frac{2p(1-p)}{(1-2p)^2} \left(\left(\frac{1-p}{p} \right)^n - \left(\frac{1-p}{p} \right)^k \right) - \frac{n-k}{1-2p}.$$

Definition 23 (Widgets). A Widget W is a tree $\mathcal{T} = (\mathcal{V}, \mathcal{E}, b)$ where $b \in \mathcal{V}$ is called the bridge. We consider the three widgets described in Figure 8.a: head, gate and tail and the three configurations $\hat{c}_l, \hat{c}_u, \hat{c}_s$ for gates.

Definition 24 (Biased trees). Let $(W_i)_{0 \leq i \leq n+1}$ be a finite sequence of widgets where $W_i = (\mathcal{V}_i, \mathcal{E}_i, b_i)$. From this sequence, we define the tree $\mathcal{T} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \cup_{i=0}^{n+1} \mathcal{V}_i$ and $\mathcal{E} = (\cup_{i=0}^{n+1} \mathcal{E}_i) \cup (\cup_{i=0}^n b_i b_{i+1})$. Abusively we also denote by $(W_i)_{0 \leq i \leq n+1}$ the tree generated by this sequence. A biased tree of size n is a finite sequence of widgets $(W_i)_{0 \leq i \leq n+1}$ where W_0 is a head, for $1 \leq i \leq n$, W_i is a gate and W_{n+1} is a tail.

Definition 25 (Stable and unstable gates). Consider a biased tree $(W_i)_{0 \leq i \leq n+1}$ and a configuration \hat{c} . We denote by \hat{c}_{W_i} , the restriction of \hat{c} to widget W_i . We say that gate i is locked if $\hat{c}_{W_i} = \hat{c}_l$, unlocked if $\hat{c}_{W_i} = \hat{c}_u$ and stable if $\hat{c}_{W_i} = \hat{c}_s$. An unstable gate is a gate which is locked or unlocked.

Definition 26 (Correct configuration). Configuration \hat{c} is correct if vertices of the head are black, the tail is white, and there exists a j such that for all $1 \leq i \leq j$ gate i is stable and for all $j < k \leq n$ gate k is unstable. We say that configuration \hat{c} is on position j . We denote by $\text{Pos}(\hat{c})$ the position of configuration \hat{c} . The position is unlocked if $j = n$ or gate $j+1$ is unlocked, the position is locked otherwise.



8.a – The 3 widgets used in the construction of a biased tree. Gray denotes the fact that the vertex state is not represented.

8.b – The three configurations \hat{c}_l, \hat{c}_u and \hat{c}_s .

Fig. 8. Widgets used in the construction of a biased tree.

Fact 27 (Active vertices) Consider a correct configuration \hat{c} on position j . The active vertices of \hat{c} are:

- Vertex b_j if $j \neq 0$.
- Vertex b_{j+1} if $j \neq n$ and gate W_{j+1} is unlocked.
- Vertex f_i if $j < i \leq n$.
- Vertex b_{n+1} if $j = n$.

Proof. Consider a correct configuration \hat{c} on position j . The only vertices which may be active in \hat{c} are vertices b_i and f_i for $1 \leq i \leq n$ and vertex b_{n+1} . Vertex b_{n+1} is active if and only if $\hat{c}(b_n) = 1$ that is to say gate W_n is stable. For all $1 \leq i \leq n$, vertex f_i is active if and only if the gate W_i is unstable that is to say $j < i \leq n$. For all $1 \leq i \leq n$, vertex b_i is inactive if $\hat{c}(b_{i-1}) = \hat{c}(b_i) = \hat{c}(b_{i+1})$. Thus among vertices $(b_i)_{1 \leq i \leq n}$, only vertices b_j and b_{j+1} may be active: vertex b_j is active and vertex b_{j+1} is active if gate W_{j+1} is unlocked. \square

Definition 28 (Final configuration). The final configuration \hat{c}_f is the configurations where vertices of the head are black, the tail is black and every gate is stable. We say that \hat{c}_f is on position $n+1$, $\text{Pos}(\hat{c}_f) = n+1$.

Lemma 29. Configuration \hat{c}_f is stable.

Proof. Consider the correct configuration \hat{c} on position n . According to fact 27, only vertices b_n and b_{n+1} are active. If vertex b_{n+1} fires, these two vertices become inactive and no other vertex becomes active. Firing vertex b_{n+1} leads to configuration \hat{c}_f . Thus \hat{c}_f is stable. \square

Corollary 30. Consider a correct configuration \hat{c} , then configuration $\hat{c}' = \hat{\delta}(\hat{c})$ is either correct or \hat{c}_f . Moreover $|\text{Pos}(\hat{c}') - \text{Pos}(\hat{c})| \leq 1$.

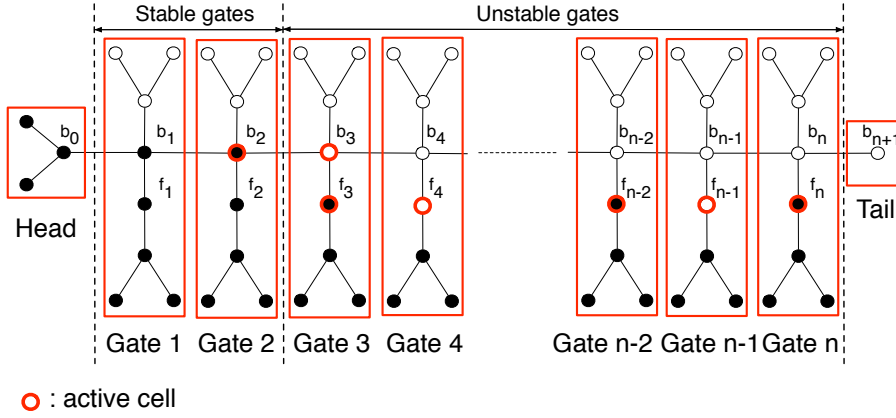


Fig. 9. A biased tree and a correct configuration on unlocked position 2.

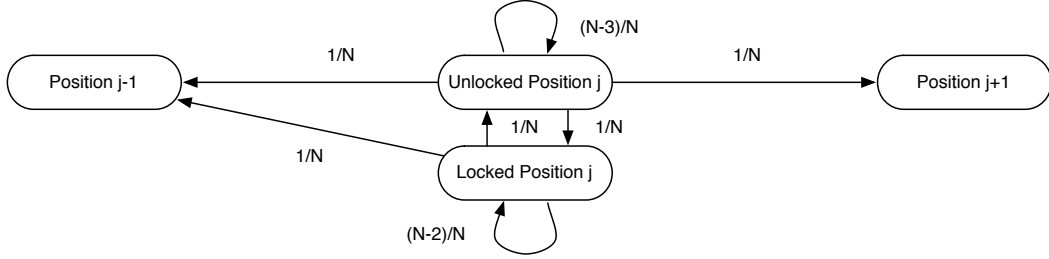
Proof. Consider a correct configuration \hat{c} on position j and the configuration $\hat{c}' = \hat{\delta}(\hat{c})$. If an inactive vertex fires then $\hat{c}' = \hat{c}$. Now consider that an active vertex fires (see fact 27):

- if $j \neq 0$ and vertex b_j fires: then gate W_j becomes unlocked and \hat{c}' is a correct configuration on unlocked position $j - 1$.
- if $j \neq n$ and vertex b_{j+1} fires: then gate W_{j+1} becomes stable and \hat{c}' is a correct configuration on position $j + 1$.
- if vertex f_i fires with $j < i \leq n$: then gate W_i becomes unlocked (resp. locked) in \hat{c}' if it is locked (resp. unlocked) in \hat{c} . Configuration \hat{c}' stays correct and on position j .
- if $j = n$ and vertex b_{n+1} fires: then $\hat{c}' = \hat{c}_f$. □

Theorem 31. *On biased trees of size n (i.e. $N = 8n + 4$ vertices), starting from a correct configuration, Stochastic Minority converges almost surely to c^f . Moreover the hitting time T of the limit set satisfies $\Theta(1.5^n) \leq \mathbb{E}(T) \leq \Theta(n4^n)$.*

Proof. Consider a biased tree of size n , an initial correct configuration \hat{c}^0 on position 0 and the sequence $(\hat{c}^t)_{t \geq 0}$. Dynamics $\hat{\delta}$ converges almost surely from initial configuration \hat{c}^0 and $c^T = c_f$. We define the sequence of random variable $(t_i)_{i \geq 0}$ as $t_0 = 0$ and $t_{i+1} = \min \{t > t_i \mid \text{Pos}(\hat{c}^{t_{i+1}}) \neq \text{Pos}(\hat{c}^{t_i}) \text{ or } \text{Pos}(\hat{c}^{t_{i+1}}) = n + 1\}$. Consider the sequence of random variable $(X_i)_{i \geq 0}$ such that $X_i = \text{Pos}(\hat{c}^{t_i})$. According to corollary 30, $|X_{i-1} - X_i| = 1$.

Consider a configuration \hat{c}^t on locked position $n > j > 0$ then firing vertex b_j leads to a configuration on position $j - 1$ and firing vertex f_j leads to a configuration on unlocked position j . Firing other vertices does not affect the position of the configuration. Consider a configuration \hat{c}^t on unlocked position $n > j > 0$ then firing vertex b_j leads to a configuration on position $j - 1$, firing vertex f_j leads to a configuration on locked position j and firing vertex b_{j+1} leads to a configuration on position $j + 1$. Firing other vertices does not affect the position of the configuration. A vertex has a probability $1/N$ to fire where $N = 4 + 8n$. Thus, the evolution of a configuration on position $0 < j < n$ can be summarized as:



A basic analysis yields that:

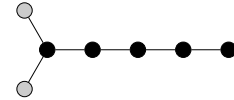
- if $1 \leq x \leq n$ then $\mathbb{P}(X_{i+1} = x + 1 | X_i = x) = 1 - \mathbb{P}(X_{i+1} = x - 1 | X_i = x)$ and $1/5 \leq \mathbb{P}(x_{i+1} = x + 1 | X_i = x) \leq 2/5$.
- $\mathbb{P}(X_{i+1} = 1 | X_i = 0) = 1$.
- $\mathbb{P}(X_{i+1} = n + 1 | X_i = n + 1) = 1$.

Thus the behavior of $(X_i)_{i \geq 0}$ is as described in definition 21. We define the random variable $T' = \min \{i \geq 0 \mid X_i = n\}$ which corresponds to the first time when all gates are stable, then $\Theta(\frac{3}{2}^n) \leq \mathbb{E}(T') \leq \Theta(4^n)$ (see Def. 21). We call c^{f-1} the correct configuration on position n (i.e. all gates are stable). Then $c^T = c^f$, $c^{T-1} = c^{f-1}$ and $\mathbb{P}(c^{t+1} = c^f \mid c^t = c^{f-1}) = 1/2$. Thus, $\mathbb{E}(T) = \Theta(\mathbb{E}(t_{T'}))$. By definition, $t_{T'} = \sum_{i=1}^{T'} (t_i - t_{i-1}) = \sum_{i=1}^{\infty} [(t_i - t_{i-1}) 1_{t_i < T'}]$. Since there are at most 2 vertices which may modify the position of a correct configuration, we have $1 \leq \mathbb{E}(t_{i+1} - t_{i-1}) \leq \Theta(n)$. Thus $\sum_{i=1}^{\infty} (1_{t_i < T'}) \leq \mathbb{E}(t_{T'}) \leq \Theta(\sum_{i=1}^{\infty} (n 1_{t_i < T'}))$. We conclude that $\Theta((\frac{3}{2})^n) \leq \mathbb{E}(T) \leq \Theta(n 4^n)$. \square

Subcase: Binary Trees Converge Rapidly In this section, we note that on binary trees, i.e. trees where the degrees are at most 3, the dynamics ends by fixing the states of the vertices of degree 1 and 3 (see Remark 10) and some isolated particles may remain and oscillate on disjoint paths.

Definition 32 (Path).

In this subcase, we call path a connected subgraph where all nodes but the end nodes have degree 2 (end nodes must thus have degree 1 or 3). An example is composed of the black nodes on this figure:



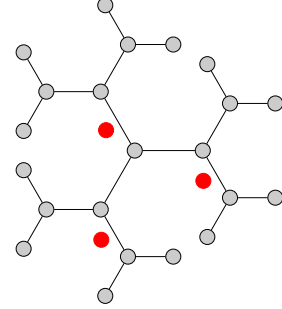
Theorem 33. Stochastic Minority on trees with degrees at most 3 hits the limit set in $O(N^4)$ steps on expectation. The attractors of a tree \mathcal{T} are in bijection with the matchings of the reduced tree \mathcal{T}' where each path of \mathcal{T} has been replaced by an edge, then each leaf has been removed.

Proof. We study here the movements of particles in the initial tree \mathcal{T} . One can divide \mathcal{T} into its induced subgraphs which are paths. Those paths link the vertices of odd degree (1 or 3). The reduced tree \mathcal{T}' is obtained by replacing each path by an edge, then removing the leaves.

Consider a configuration c on \mathcal{T} which belongs to an attractor. There cannot be two particles on the same path, otherwise a sequence of updates could lead to the collision of

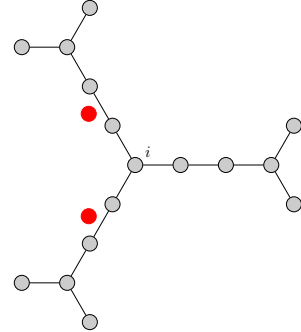
these two particles and thus to an energy decrease. In the same way, there cannot be two particles on two paths which share a common extremity i . This extremity would necessarily be a vertex of degree 3, then a sequence of updates could position the two particles on the edges incident to i . Firing i at that time would decrease the number of particles by at least 1 and lead to an energy decrease. Finally, there cannot be a particle on a path which has an end of degree one, since the particle could disappear at this end.

Reciprocally, for configurations where there are no particles on the same path nor on adjacent paths nor on a path with an end of degree 1, the number of particles on each path is constant. This proves that the energy cannot decrease, and that such configurations belong to the limit set. This also establishes the bijection between attractors and matchings of \mathcal{T}' (the matching indicates where the isolated particles are located in \mathcal{T}). An illustration of \mathcal{T}' is on the opposite figure, particles are next to the edges.



To prove the bound on the expected hitting time of the limit set, we find a bound on the time until at least one particle disappears. Consider a configuration where there exist two particles on a same path of length n . One can suppose that these two particles follow a random walk on this path with reflecting barriers at each extremity, unless they collide with another particle (leading to the loss of two particles in the tree) or unless one of the two particles leaves the path (leading to the loss of at least one particle in the tree since leaving necessarily involves a vertex of degree 3 fired with two incident particles). Thus under the condition that they have not disappeared before, a bound on the expected time elapsed until they collide can be derived from classical studies of random walks with reflecting barriers [39]: this expected time is bounded by $O(n^3)$, and consequently it is also a bound on the time until at least one particle disappear.

Consider a configuration where there exist two particles on two paths of respective lengths n and m , sharing a common extremity: the vertex i . An illustration is on the opposite figure. With the same reasoning, assuming that the two particles have not led to the removal of another particle means that they follow random walks on their respective paths with reflecting barriers at the extremities. Then they can only disappear by being both incident to vertex i when vertex i is fired. By analyzing the two-dimensional random walk corresponding to the evolution of the respective distances to vertex i , this event occurs after at most $O(\max(n, m)^3)$ steps on expectation, as proved in Section B (or in [41]) using standard tools for multi-dimensional random walks.



Finally, from any configuration which does not belong to the limit set, at least one particle disappear within $O(N^3)$ steps on expectation. Since the number of particles in any initial configuration on a tree is bounded by N , the expected time to hit the limit set is bounded by $O(N^4)$. \square

4.3 Phase transition

In this part, we consider the infinite graph where the set of vertices is \mathbb{N} and vertex i has two neighbors: $i - 1$ and $i + 1$. We consider the initial configuration c^{init} where the state of vertex i is 1 if $i = 0$ and $i \bmod 2$ otherwise. This configuration possesses only three vertices which are not in their minority state: -1 , 0 and 1 . Updating vertex 0 leads to a stable configuration.

We consider the fully asynchronous dynamics: at each time step, only one vertex is updated and this vertex is selected uniformly at random among the active vertices. Note that the set of active vertices is always finite, so this random selection is feasible. The limit set of an execution of Minority starting from the initial configuration c^{init} is composed of the stable configuration where the state of vertex i is $i \bmod 2$.

We denote by $\mathbb{P}_\alpha(c^{init})$ the probability that the dynamics reaches a stable configuration from the initial configuration c^{init} . If $\mathbb{P}_\alpha(c^{init}) < 1$ then the expected time to reach the limit set is infinite. Experimentally, there is a phase transition on $\mathbb{P}_\alpha(c^{init})$ depending on α with a critical value $\alpha_c \approx 0.5$ such that:

- if $\alpha < \alpha_c$ then $\mathbb{P}_\alpha(c^{init}) = 1$;
- if $\alpha > \alpha_c$ then $\mathbb{P}_\alpha(c^{init}) < 1$

Our result link the critical value α_c to the critical value $0.6298 < p_c < 2/3$ of directed percolation.

Theorem 34. *If $\alpha \geq \sqrt[3]{1 - (1 - p_c)^4}$ then $\mathbb{P}_\alpha(c^{init}) < 1$. The quantity $\sqrt[3]{1 - (1 - p_c)^4}$ is in $[0.993; 0.996]$; thus if $\alpha \geq 0.996$ then $\mathbb{P}_\alpha(c^{init}) < 1$.*

Proof. In [37], this theorem is proved for this dynamics with the rule FLIP-IF-NOT-EQUAL (an updated vertex switches its state if at least one of its neighbor is in a different state than itself) from the initial configuration c'^{init} where vertex i is in state 0 if $i \neq 0$ and 1 otherwise. We simply prove here that FLIP-IF-NOT-EQUAL is the dual dynamics of Minority obtained by switching state of vertices i if $i \bmod 2 = 1$:

- if vertex i is active for Minority then at least one of its neighbors is in the same state as itself. Thus in the dual configuration, vertex i has at least one of its neighbors in a different state as itself.
- if vertex i is inactive for Minority then both its neighbors are in a different state than itself. Thus, in the dual configuration, vertex i has both its neighbors in the same state as itself.

Also, c'^{init} is the dual configuration of c^{init} .

5 Conclusion

The table below sums up the different worst case average hitting times of a limit set for different topologies and compares the fully asynchronous dynamics to the synchronous one. In the case of the torus under fully asynchronous dynamics, it is conjectured that this average “convergence” time admits a polynomial bound in the number of cells.

	Fully Asynchronous	Synchronous
Path or Cycle	Poly	Exp [33]
Tree, max degree ≤ 3	Poly	Exp [33]
Tree, max degree ≥ 4	Exp	Exp [33]
Torus, von Neumann neighborhood	Poly ? [1]	Exp [33]
Torus, Moore neighborhood	Poly ? [2]	Exp [33]
Clique	Poly	Poly [33]

The Minority rule admits a rich range of behaviors under full asynchronism. The case of trees has shown that the average hitting time of limit sets is not necessarily polynomial under full asynchronism (there is a threshold on the maximum degree). For now, it is an open problem to predict from the graph topology whether the dynamics will converge fast or slowly. Following this work, a challenge is to identify the graph parameters that guide this convergence speed, as well as extending such results to other updating rules.

Acknowledgements. We are very grateful to the referees for their comments and suggestions which have improved the paper.

References

1. Regnault, D., Schabanel, N., Thierry, E.: Progresses in the analysis of stochastic 2D cellular automata: A study of asynchronous 2D minority. *Theoretical Computer Science* **410** (47-49) (2009) 4844–4855
2. Regnault, D., Schabanel, N., Thierry, E.: On the analysis of "simple" 2D cellular automata. *Discrete Mathematics and Theoretical Computer Science (DMTCS)* **12**(2) (2010) 263–294 online publication, <http://www.dmtcs.org/dmtcs-ojs/index.php/dmtcs/article/view/1337>.
3. Kittock, J.E.: Emergent conventions and the structure of multi-agent systems. In: *Proceedings of the 1993 Santa Fe Institute Complex Systems Summer School*. (1993)
4. Dyer, M., Goldberg, L.A., Greenhill, C., Istrate, G., Jerrum, M.: Convergence of the iterated prisoner's dilemma game. *Combinatorics, Probability and Computing* **11** (2002) 125–147
5. Mossel, E., Roch, S.: Slow emergence of cooperation for win-stay lose-shift on trees. *Machine Learning* **67**(1-2) (2006) 7–22
6. Kerr, B., Riley, M.A., Feldman, M.W., Bohannan, B.J.M.: Local dispersal promotes biodiversity in a real-life game of rock-paper-scissors. *Nature* **418** (2002) 171–174
7. Ising, E.: Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik* **31** (1925) 253–258
8. Srivastava, R., Ashok, J.: *Statistical Mechanics*. Prentice-Hall of India (2005)
9. McCoy, B., Wu, T.T.: *The two-dimensional Ising model*. Harvard University Press (1974)
10. Bodini, O., Fernique, T., Regnault, D.: Quasicrystallization by stochastic flips. *Proceedings of Aperiodic 09* (2009) to appear
11. Bodini, O., Fernique, T., Regnault, D.: Stochastic flips on two-letter words. In: *Proceedings of the Seventh Workshop on Analytic Algorithmics and Combinatorics (ANALCO 2010)*, SIAM (2010) pages 48–55, online publication <http://www.siam.org/proceedings/analco/2010/analco10.php>.
12. Gács, P.: Reliable computation with cellular automata. *Journal of Computer and System Sciences* **32**(1) (1986) 15–78
13. Gács, P.: Reliable cellular automata with self-organization. *Journal of Statistical Physics* **103**(1/2) (2001) 45–267

14. Gács, P., Reif, J.: A simple three-dimensional real-time reliable cellular array. *Journal of Computer and System Sciences* **36**(2) (1988) 125–147
15. Toom, A.: Nonergodic multidimensional systems of automata. *Problems of Information Transmission* **10** (1974) 239–246
16. Toom, A.: Stable and attractive trajectories in multicomponent systems. *Advances in Probability* **6** (1980) 549–575
17. Toom, A., Vasilyev, N., Stavskaya, O., Mityushin, L., Kurdyumov, G., Pirogov, S.: Discrete local Markov systems. In: *Stochastic cellular systems: ergodicity, memory, morphogenesis*. Manchester University Press (1990) 1–182
18. Balister, P., Bollobás, B., Kozma, R.: Large deviations for mean fields models of probabilistic cellular automata. *Random Structures & Algorithms* **29** (2006) 399–415
19. Buvel, R., Ingerson, T.: Structure in asynchronous cellular automata. *Physica D* **1** (1984) 59–68
20. Huberman, B.A., Glance, N.: Evolutionary games and computer simulations. *Proceedings of the National Academy of Sciences, USA* **90** (Aug. 1993) 7716–7718
21. Bersini, H., Detours, V.: Asynchrony induces stability in cellular automata based models. In: *Proceedings of Artificial Life IV*, Cambridge, MIT Press (1994) 382–387
22. Kanada, Y.: Asynchronous 1D cellular automata and the effects of fluctuation and randomness. In: *Proceedings of the fourth conference on artificial life (A-Life IV)*, MIT Press (1994)
23. Schönfisch, B., de Roos, A.: Synchronous and asynchronous updating in cellular automata. *Biosystems* **51**(3) (1999) 123–143
24. Fatès, N., Morvan, M., Schabanel, N., Thierry, E.: Fully asynchronous behavior of double-quiescent elementary cellular automata. *Theoretical Computer Science* **362** (2006) 1–16
25. Fatès, N., Regnault, D., Schabanel, N., Thierry, E.: Asynchronous behavior of double-quiescent elementary cellular automata. In Correa, J.R., Hevia, A., Kiwi, M.A., eds.: *LATIN Proceedings*. Volume 3887 of *Lecture Notes in Computer Science*, Springer (2006) 455–466
26. Fukás, H.: Probabilistic cellular automata with conserved quantities. *Nonlinearity* **17**(1) (2004) 159–173
27. Fatès, N., Morvan, M.: Perturbing the topology of the game of life increases its robustness to asynchrony. In Sloot, P.M.A., Chopard, B., Hoekstra, A.G., eds.: *6th International Conference on Cellular Automata for Research and Industry Proceedings (ACRI)*. Volume 3305 of *LNCS*, Springer (Oct. 2004) 111–120
28. Fatès, N., Morvan, M.: An experimental study of robustness to asynchronism for elementary cellular automata. *Complex Systems* **16**(1) (2005) 1–27
29. Rouquier, J.B., Morvan, M.: Combined effect of topology and synchronism perturbation on cellular automata: Preliminary results. In Umeo, H., Morishita, S., Nishinari, K., eds.: *ACRI*. Volume 5191 of *LNCS*, Springer Berlin (2008)
30. Ermentrout, G.B., Edelstein-Keshet, L.: Cellular automata approaches to biological modelling. *Journal of Theoretical Biology* **160** (1993) 97–133
31. Silva, H.S., Martins, M.L.: A cellular automata model for cell differentiation. *Physica A: Statistical Mechanics and its Applications* **322** (2003) 555–566
32. Liggett, T.M.: *Interacting Particle Systems, an Introduction*. Springer Verlag (2004)
33. Goles, E., Martinez, S.: *Neural and automata networks, dynamical behavior and applications*. Volume 58 of *Maths and Applications*. Kluwer Academic Publishers (1990)
34. Istrail, S.: Statistical mechanics, three-dimensionality and NP-completeness: I. universality of intractability for the partition function of the ising model across non-planar lattices. In: *Proceedings of STOC*, ACM Press (2000) 87–96
35. Rojas, R.: *Neural networks: a systematic introduction*. Springer (1996) Chap. 13 - The Hopfield Model.
36. Rouquier, J.B., Morvan, M.: Coalescing cellular automata: Synchronization by common random source for asynchronous updating. *Journal of Cellular Automata* **4** (2009) 55–78

37. Regnault, D.: Directed percolation arising in stochastic cellular automata analysis. In: Proceedings of the 33rd International Symposium on Mathematical Foundations of Computer Science. Volume 5162 of LNCS. (2008) 563–574
38. Lindvall, T.: Lectures on the coupling method. Wiley (1992)
39. Grimmett, G., Stirzaker, D.: Probability and random process. 3rd edn. Oxford University Press (2001)
40. Grigoriev, S., Priezzhev, V.: Random walk of annihilating particles on the ring. Theoretical and Mathematical Physics **146**(3) (2006) 411–420
41. Regnault, D., Rouquier, J.B., Thierry, E.: Stochastic minority on graphs. Technical report, LIP, ENS Lyon, Université de Lyon (April 2008)
42. Levin, D.A., Peres, Y., Wilmer, E.L.: Markov chains and mixing times. AMS (2009)

A Characterization of stable configurations on trees

Let us now study more precisely the structure of the limit set. This characterization allows to count the number of attractors on a given tree.

We first pick an arbitrary vertex of degree 1 and set it as root r in the tree (this introduces “parent” and “children” relations). The term “degree” refers to the original graph: a vertex of degree three has two children.

We assign a label to each vertex to count the number of attractors and the size of the limit set. The number of attractors will be shown to be the number of acceptable labelings. The set of labels we use is $\{(\square, 0), (\square, 1), (\triangleright, 0), (\triangleright, 1)\}$. “ \triangleright ” intuitively means “oscillating” while “ \square ” means “fixed” (like the recorder symbols play/stop). The second component is called the “preferred” state of the vertex.

Definition 35 (acceptable labeling). *A labeling is acceptable if and only if, for each vertex i , if the vertex has label*

1. (\square, α) then it has strictly more than $\deg(i)/2$ neighbors with label (\square, α) ;
2. (\triangleright, α) then
 - (a) if the parent has a label of the form (\square, β) , then $\alpha = 0$ and i has one more child labeled $(\cdot, 1 - \beta)$ than children labeled (\cdot, β) ;
 - (b) otherwise, i.e. the parent has a label of the form (\triangleright, \cdot) , i has one more child labeled (\cdot, α) than children labeled $(\cdot, 1 - \alpha)$.

Note that only vertices of even degree can have a label of the form (\triangleright, \cdot) . The apparent asymmetry in case 2a (imposing $\alpha = 0$) is there only to avoid double counts in Theorem 37, one could as well have defined acceptable labelings with $\alpha = 1$.

Theorem 37 shows that a labeling corresponds to an attractor, and Theorem 38 details the meaning of a labeling, thus the structure of an attractor.

For a labeling L , we define a few special configurations: $\text{snd}(L)$ is the projection of the second component: $L_i = (\cdot, \alpha) \iff (\text{snd}(L))_i = \alpha$, and $\text{paint}(L, \alpha)$ sets all vertices labeled (\triangleright, \cdot) to α :

$$(\text{paint}(L, \alpha))_i := \begin{cases} \beta & \text{if } L_i = (\square, \beta) \\ \alpha & \text{if } L_i = (\triangleright, \cdot) \end{cases}$$

For a configuration c in the limit set, we denote $\text{attr}(c)$ the attractor containing c .

Lemma 36. *If L is an acceptable labeling, $\text{snd}(L)$ and $\text{paint}(L, \alpha)$ are in the limit set.*

Proof. Consider the configuration $\text{paint}(L, 0)$. We consider the vertices in a bottom-up order (a vertex is considered after all its children), and update each vertex which is not in its preferred state. Updating a vertex makes it go to its preferred state thanks to the definition of acceptable labelings. The current configuration is $\text{snd}(L)$.

Then, we consider the vertices in a top-down order and update those that are in state 0. When a vertex is considered, its children are in their preferred state, and if the parent of a vertex is labeled (\triangleright, \cdot) then this parent is in state 1. Again, thanks to the definition of acceptable labelings, updating a vertex makes it go to state 1. We get the configuration $\text{paint}(L, 1)$.

By symmetry, there is a sequence of updates from $\text{paint}(L, 1)$ to $\text{paint}(L, 0)$. Moreover, there is no sequence of updates leading from one of this configurations to change of state of a vertex labeled (\square, α) : the first change of state of such a vertex i would contradict the fact that it has more than $\deg(i)/2$ neighbors with label (\square, α) and thus in the state α . Which implies that, whatever the configuration reached from $\text{paint}(L, 0)$, there is by monotonicity a sequence of updates yielding the configuration $\text{paint}(L, 1)$. The full cycle is in the limit set. \square

Theorem 37. *Given a tree, there is a bijection between attractors and acceptable labelings.*

Proof. We define a function f that maps an attractor A to an acceptable labeling L . We then shows that for every attractor A , $\text{attr} \circ \text{snd}(f(A)) = A$ and for every acceptable labeling L , $f(\text{attr} \circ \text{snd}(L)) = L$. This imply that f is a bijection.

To define f , let A be an attractor, we construct $L = f(A)$. For all vertices that have the same state α in any configuration of A , define $L_i := (\square, \alpha)$. All leaves are now labeled. We define the labeling of the remaining vertices inductively (in a bottom-up order). Each remaining vertex is oscillating, thus has an even degree and an odd number of children. Considering a vertex i having all its children labeled:

- if the parent is already labeled (thus with a label of the form (\square, \cdot)), define $L_i := (\triangleright, 0)$;
- otherwise, let α be the majority state among the preferred states of the children and define $L_i := (\triangleright, \alpha)$. (Since i has an odd number of children, the majority state is well defined).

Let us show that the labeling we have just defined is acceptable. With the same argument as the proof of proposition 17, the configuration where all oscillating vertices are in state α is in A . This configuration is $\text{paint}(L, \alpha)$.

1. Consider a vertex i labeled $(\square, 0)$ and the configuration $\text{paint}(L, 1)$. All the neighbors of i not labeled $(\square, 0)$ are in the state 1. So, there are necessarily more than $\deg(i)/2$ neighbors labeled $(\square, 0)$ (and thus in the state 0), otherwise, updating i would make it change its state, contradicting the definition of $L_i = (\square, \cdot)$. By symmetry, point 1 of the definition of acceptable labelings is fulfilled.

2. We show point 2 inductively in a bottom-up order. Consider a vertex i labeled (\triangleright, α) . Let us first show that there is a configuration $c \in A$ where all the vertices of the subtree having i as root are in their preferred state, except i (which is in state $1 - \alpha$). Indeed, consider $\text{paint}(L, 1 - \alpha)$ and update the vertices not in their preferred state in a bottom-up order. Thanks to the induction hypothesis, the label of those vertices is acceptable, thus updating them makes them change to their preferred state. This is a sequence of updates from a configuration in A leading to c , so c belongs to A . We can now show that the label of i is acceptable:
- (a) If the parent has a label of the form (\square, \cdot) then L_i has been defined as $(\triangleright, 0)$. Moreover, in c , the parent of i is in state β . In this configuration, updating i cannot make the energy decrease, so i must have one more child labeled $(\cdot, 1 - \beta)$ than children labeled (\cdot, β) . So, point 2a is fulfilled.
 - (b) Otherwise, the parent is labeled (\cdot, β) . From the definition of L , in configuration c , i has a majority of children in state α . Updating it thus makes it change its state. So i must have as many neighbors in state α than in state $1 - \alpha$. Which means that $\beta = 1 - \alpha$ and i has exactly one more child labeled (\cdot, α) than children labeled $(\cdot, 1 - \alpha)$. Point 2b is fulfilled.

Let us now show that for any acceptable labeling L' , $f(\text{attr} \circ \text{snd}(L')) = L'$. Since $\text{paint}(L', 0)$ and $\text{paint}(L', 1)$ are in the attractor $\text{attr} \circ \text{snd}(L')$ (cf. Lemma 36), the vertices labeled (\triangleright, \cdot) in L' are oscillating in this attractor. From the definition of f , these vertices are labeled (\triangleright, \cdot) in $f(\text{attr} \circ \text{snd}(L'))$.

Recall that there is no sequence of updates leading from $\text{snd}(L')$ to change the state of a vertex labeled (\square, α) : the first change of state of such a vertex i would contradict the fact that it has more than $\deg(i)/2$ neighbors with label (\square, α) and thus in the state α . This allows us to conclude that the labelings L' and $f(\text{attr} \circ \text{snd}(L'))$ have the same cells labeled $(\square, 0)$ and $(\square, 1)$, thus they are equal. (Indeed, in the definition of f , the value of α for vertices labeled (\triangleright, α) is entirely determined by the labeling of vertices labeled (\square, \cdot) .)

Finally, let A' be an attractor, we show that $\text{attr} \circ \text{snd}(f(A')) = A'$. $f(A')$ is an acceptable labeling so (Lemma 36), $\text{paint}(f(A'), 1)$ is in the attractor $\text{attr} \circ \text{snd}(f(A'))$. Moreover, we have already noted that the configuration where all the oscillating vertices of A' are in state 1 belongs to A' (same argument as the proof of Proposition 17). From the definition of f , this configuration is $\text{paint}(f(A'), 1)$. The attractors $\text{attr} \circ \text{snd}(f(A'))$ and A' have the element $\text{paint}(f(A'), 1)$ in common, so they are the same attractor.

This concludes the proof by implying that f is a bijection. \square

Theorem 38 (Structure of an attractor). *Let L be an acceptable labeling. Then for every configuration c reachable by a sequence of updates from $\text{snd}(L)$, for every vertex i :*

1. If $L_i = (\square, \alpha)$ then $c_i = \alpha$ (this is why “ \square ” intuitively means “fixed”).
2. If $L_i = (\triangleright, \alpha)$ (in this case $\deg(i)$ is even: i is not the root, which has degree 1) then
 - (a) if the parent has a label of the form (\square, \cdot) , i is in the state appearing in majority among its neighbors (no constraint in case of equality);
 - (b) otherwise

- if i is in its preferred state α , its children labeled (\cdot, α) are in their preferred state α
- otherwise, all its children not in their preferred state are in the same state as i (the state $1 - \alpha$).

Proof. Recall that there is no sequence of updates from $\text{snd}(L)$ to the firing of a vertex labeled (\square, α) . This ensures point 1.

Since $\text{snd}(L)$, thus c , are in the limit set, the energy cannot decrease. It follows that any vertex i has always at least $\deg(i)/2$ neighbors in the same state as i . This proves point 2a.

We prove the last point (2b) by recursion. Note that the assertion of the theorem is clearly true for $\text{snd}(L)$. Let c be a configuration verifying the assertion, c' a configuration reached from c by updating a vertex i , it is sufficient to prove that the assertion is true for c' . Precisely, we prove that it is true for i and its neighbors.

For i :

- If i is in its preferred state α in c , then its children labeled (\cdot, α) (there are $\deg(i)/2$ such children thanks to the definition of acceptable labelings) are then in state α in c and thus in c' . Which means that i can change its state only if all other children are in state $1 - \alpha$. Point 2b stays true for i in c' .
- Otherwise, it is either inactive and point 2b stays true in c' , or active. In the latter case, i has as many neighbors in each state (because c is in the limit state), all children of i not in their preferred state are in the same state as i (by recursion hypothesis) and i has one more child labeled (\cdot, α) than children labeled $(\cdot, 1 - \alpha)$ (definition of acceptable labelings). These three conditions imply that all children of i are actually in their preferred state. So, c' fulfills point 2b.

For a neighbor w of i :

- If w is a child of i , nothing has changed for the sons of w : point 2b stays true.
- If w is the parent of i , it is sufficient to consider the case where i is in the same state α as w (there is no condition to check in the other case).
 - If α is the preferred state of i then i has at least $\deg(i)/2 + 1$ neighbors in state α : its children labeled (\cdot, α) and its parent w . So i is inactive.
 - Otherwise, i and w are not in their preferred state, and point 2b stays true, whether or not i change its state. \square

B Omitted proofs: Bound on the Hitting Time of a 2D Finite Markov Chain

B.1 Background on Markov chain theory

We recall only the necessary background on Markov chains to get a bound on the hitting time of a 2D finite Markov chain. For a gentle introduction and proofs, we refer for instance to Chapter 10 of [42].

Let $(X_t)_{t \in \mathbb{N}}$ be a Markov chain. We note τ_b the **hitting time** of b , i.e. the first time the Markov chain is in state b :

$$\tau_b := \min \{t \geq 0 \mid X_t = b\} .$$

If (X, P) is a Markov chain reversible with respect to the probability π , the **conductance** of an unoriented edge (x, y) is

$$g(x, y) := \pi(x)P(x, y) = \pi(y)P(y, x) .$$

Let a and b be two distinguished vertices, representing source and sink. We use the following potential, or **voltage**, of a vertex:

$$V(x) := \mathbb{P}(\tau_a < \tau_b \mid X_0 = x) .$$

Clearly $V(a) = 1$ and $V(b) = 0$. Now, define the **current flow** on oriented edges as

$$I(x, y) := g(x, y) (V(x) - V(y)) \quad \text{and} \quad \|I\| := \sum_x I(a, x) .$$

The **effective resistance** between a and b is

$$\mathcal{R}(a, b) := \frac{V(a) - V(b)}{\|I\|} .$$

Theorem 39 (Commutate time identity).

$$\mathbb{E}(\tau_b \mid X_0 = a) + \mathbb{E}(\tau_a \mid X_0 = b) = g\mathcal{R}(a, b)$$

In our case, $g = 1$, so $\mathcal{R}(a, b)$ is an upper bound for the average hitting time $\mathbb{E}(\tau_b \mid X_0 = a)$. Here is how one can bound $\mathcal{R}(a, b)$. A **flow** from a to b is a function on oriented edges which is antisymmetric: $\theta(x, y) = -\theta(y, x)$ and which obeys Kirchhoff's vertex law:

$$\forall v \notin \{a, b\} \quad \sum_x \theta(x, v) = 0 .$$

This is just the requirement “flow in equals flow out”. The **strength** of a flow is

$$\|\theta\| := \sum_x \theta(a, x) .$$

Theorem 40 (Thomson's Principle). *For any finite connected graph,*

$$\mathcal{R}(a, b) = \inf \{E(\theta) \mid \theta \text{ a unit flow from } a \text{ to } b\}$$

$$\text{where } E(\theta) := \sum_{x, y} \frac{(\theta(x, y))^2}{g(x, y)} .$$

B.2 Application to our case

If a neighbor does not exist (because the vertex is on the border), the edge points to the vertex itself. Each edge has the same probability $\frac{1}{4}$.

An invariant probability is the uniform probability $\pi : x \mapsto \frac{1}{nm}$. Our Markov chain is reversible with respect to this probability. So we can use the definitions of section B.1:

$$\forall x, y \quad g(x, y) = \frac{1}{4nm} .$$

Thanks to Thomson's principle, it is sufficient to construct a flow from a to b to get an upper bound on $\mathcal{R}(a, b)$. If $a = (i, j)$ and $b = (n, m)$, we consider the trivial (and far from optimal) flow

$$\begin{cases} \theta((k, j), (k+1, j)) := 1 & \text{if } i \leq k < n \\ \theta((n, k), (n, k+1)) := 1 & \text{if } j \leq k < m \\ \theta(x, y) := 0 & \text{elsewhere.} \end{cases}$$

That is, a flow on a single path from a to b .

$E(\theta) \leq (n+m)4nm$. We conclude with the commute time identity that the average hitting time is $O(n^3)$, assuming w.l.o.g. that $n \geq m$.